Sensor Integration and Context Detection in Intelligent Systems

Assoc. Prof. Anton Vitko, Dipl.Ing. Michal Šavel, Dipl.Ing. Dušan Kameniar

and Prof. Jurišica Ladislav*

Slovak University of Technology, Faculty of Mechanical Engineering
Námestie Slobody 17, 812 31 Bratislava, Slovakia
*Faculty of Electrical Engineering and Information Technology
Ilkovicova 3, 81219 Bratislava
SLOVAKIA

Email: vitko@kam.vm.stuba.sk

ABSTRACT

Sensor data integration into a more information-rich structure that reflects both the system state of and the state of its environment i.e. context information is a subject of the intensive study in military applications. Endowing an object with the ability of grasping context information is one of the prior prerequisite of its autonomous operability. An autonomously operating system, be it a terrestrial mobile machine or UAV, is required to respond to instantaneous incentives coming form the surrounding environment. To this end the system needs to handle wide range of unexpected contexts. In particular the system should be able to distinguish between common (normal) and unusual (abnormal) contexts. The distinguished contexts should be then classified with respect to their criticality. To perform these tasks, the system functionality must be organized into an appropriate architecture, i.e. a set of organizing principles and core components that are used to build the basis for the system. The first part of the paper summarizes both features of intelligent systems and current approaches to the sensor integration. The aim is to imbue the intelligent system with the ability to acquire the context. Within the second part the topic is narrowed and focused on the results obtained by the originally developed fusing and classification algorithm of detection and classification of abnormal behavioural contexts. The results obtained were verified by simulation as well as by experimenting with a walking machine.

1.0 FEATURES OF AN AUTONOMOUSLY OPERATING MACHINE

Autonomously operating machine needs to handle a wide range of unexpected events, detect and distinguish normal and faulty states and classify them according their criticality. To perform these tasks, the machine's functionality must be organized into an appropriate architecture, i.e. a set of organizing principles and core components that are used to build the basis for the system. The term "intelligent behaviour", is commonly related to the abilities the conventional system cannot attain. Leaving alone the general meaning of the concept, it would be useful to single out some basic features that could be used for characterizing an intelligent system. As early as at the beginning of seventies K.S. Fu [1] linked intelligent behaviour with the features that were traditionally out of the scope of specialists working in conventional control theory. These are mainly the abilities of making decisions, adapt to new and uncertain situations, self-organization, planning, image recognition, and more. [2] Intelligent systems should not be restricted to those that are based on a particular constituent of soft computing techniques (fuzzy logic, neural networks, genetic algorithms and probabilistic reasoning), as it is frequently done. Soft computing techniques should be considered as mere building blocks or even "bricks" used for building up a "large house" of an intelligent system. What makes today's systems intelligent is just a synergic use of these

Vitko, A.; Šavel, M.; Kameniar, D.; Ladislav, J. (2005) Sensor Integration and Context Detection in Intelligent Systems. In *Advanced Sensor Payloads for UAV* (pp. 4-1 – 4-10). Meeting Proceedings RTO-MP-SET-092, Paper 4. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int/abstracts.asp.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 MAY 2005	**N/A**	**-**

4. TITLE AND SUBTITLE
Sensor Integration and Context Detection in Intelligent Systems

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Slovak University of Technology, Faculty of Mechanical Engineering Námestie Slobody 17 , 812 31 Bratislava, Slovakia

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM202032., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**10**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

techniques, which in time and space invoke, optimise and fuse elementary behaviours into an overall system behaviour. For instance, fuzzy inference is a computing framework based on the fuzzy reasoning. But the fuzzy system is not able to learn; therefore a kind of neural network is used to provide its learning ability. To this end, the fuzzy rule-set is commonly arranged into a special neural architecture like ANFIS and NEFCON with Takagi-Sugeno-Kang and Mamdani inference respectively. [3] Intelligence of neuro-fuzzy systems springs from successive generalization of the information chunks (granules) from singular ones, through crisp granular, to fuzzy granular information. [4, 5] An inferential process then runs over (overlapping) information granules. Due to the information granularization a system becomes robust with respect to imprecision, uncertainties, and partial truth. Thus, the system's intelligence comes from the system architecture, i.e. an inner organization of the both system elements and functionalities. To demonstrate this, let us look at the *subsumption architecture*, developed in 1986 by Brooks [6, 7] and used also used for design of navigation algorithm [8] of our walking machine.. The subsumption architecture was inspired by the behaviour of living creatures and, it is worth saying that, it heralded a fundamentally new approach to developing more intelligent machines. The machine behaviour is here typically broken down into a set of simpler behaviours that are loosely co-ordinated towards a final goal in a sense, that every behaviours selectively assumes the control of all subsumed behaviours. Contrary to the hierarchical architecture, where a particular behaviour assumes control when a given set of logical conditions is fulfilled while taking only little interest in other behaviours, in the subsumption architecture various behaviours can appear concurrently and with different intensity. The behaviours with higher priorities are subsumed under those with lower priorities; hence a layered structure is developed. The layer (i.e. a set of behaviours of the same priority) having assigned higher priority can inhibit even supersedes those having assigned lower priorities. For example, navigating the walking machine in an unknown environment cluttered with obstacles, it is natural to assign the highest priority to the behaviour what is typical for obstacle-avoidance behaviour, and lower priorities to behaviours, which must be initialised if the machine performs any other operation, e.g. if it finds itself trapped in a deadlock and tries to escape the deadlock. Using such priority management, the walking machine when finding itself in a deadlock inhibits all obstacle-avoidance related behaviours and the behaviour what allows to escape from the deadlock assumes control. In other words, the obstacle avoidance behaviour is normally "subsumed" under the deadlock-resolving behaviour, but if the mobile machine finds that it wanders in the deadlock (for instance in a partly closed space), the obstacle-avoidance behaviour is (to some extent) inhibited by a chosen deadlock-resolving behaviour. Similarly, a striving-towards–a-goal behaviour subsumes both of them and therefore it possesses the lowest priority. An example of subsumption architecture that was used in the navigation of our experimental walking machine [8] is schematically depicted in Fig 1.

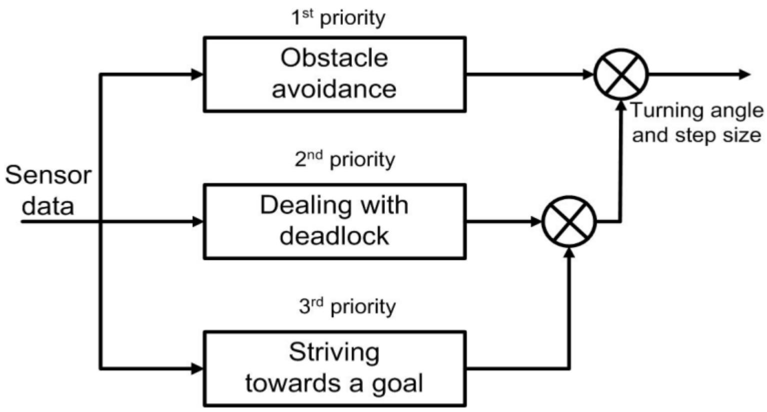


Figure 1: Subsumption architecture

One reason why the highest priority is assigned to the obstacle-avoidance behaviour is that one can reasonably expect that the machine come across an obstacle when moving in an obstacle-cluttered terrain. The deadlock-resolving behaviour (with lower priority) subsumes the previous one because it is less probable that the machine will be trapped in a deadlock. These two behaviours are subsumed by the goal-striving behaviour (with the lowest priority), because the probability that an obstacle-free landscape will appear in front of the robot is relatively low. If it actually happens, the goal striving behaviour can inhibit or even totally block both above-mentioned behaviours. Let us note that the subsumption architecture is a kind of behaviour-based architectures. [9] When implemented by fuzzy IF-THEN rules the transition between behaviours is very smooth. If the transitions are exclusively controlled by the contents of current sensor information the system belongs to the category of co called *reactive systems* [9]. The reactive systems typify the majority of autonomous machines operating in distant and unknown environments, like seabed, battlefields, areas hit by disasters etc. It would be reasonable to stress again that the system may be called intelligent mainly due to its inherent architecture. Any kind of the neuro-fuzzy learning mechanism is a mere means through which the intelligent behaviour could be implemented rather then a "source" of intelligence. The authors share the opinion that the systems that have the functionalities organized into behaviour-based agent architecture occupy the highest position in the realm of so-called intelligent systems.

2.0 BASIC ISSUES OF SENSOR INTEGRATION

Autonomously operating machine is an instantiation of the intelligent system. Its functionality strongly relies on numerous disparate sensors through which the machine grasps a consistent image of what is going on in it and around it. An underlying idea of the sensor integration rests on a synergic use of the overlapping information delivered by the sensors of different kinds. An aim is to obtain aggregated information that would be more complex then information received from a single sensor. Such blended information is beneficial at least from the aspects of noise reduction and novelty extraction. This makes the data patterns hidden in raw signals more obvious.

As a rule, a single sensor cannot provide the required amount of information. As to the mobile machine operates in changing environment the fusion must take place not only in space but also in time. Besides, using a set of sensors of different modalities offers the possibility to fuse high-level information (e.g. statements) and even to grasp a context. For instance, the fact of finding a personal mine implies a somewhat higher likelihood of finding other mines or even a whole battlefield (i.e. the context). In order to know "what to fuse", multimode information must be fused into a common format. The uncertainty of sensed and fused signals must be taken into account.

2.1 Aims and Hierarchy of Sensor Fusion

First aim is to fuse *complementary* information to obtain not only accurate but also more complete information. The adjective "complementary"indicates that data from disparate sensors are mutually complemented. For instance, images from two cameras looking in different directions are fused to obtain a more complex image. Another aim is using two or more different sensors for sensing the same quantity. For instance, sonar and laser range sensors can be used as range sensors. In this case the sensors "compete" in a sense, therefore one can speak about *competitive fusion.* [11].

The fusion runs at the different hierarchical levels. At the lowest level it is performed the signal or pixel fusion. For instance, applying logical operators may fuse pixels. If grey values of the neighbouring pixels are above given a threshold the AND filter is assumed to be true. Features are extracted at the second level. A feature is a pattern occurring in a data set that manifests correlation relationships between various components of the data. Common features are mean value, variance, covariance, power spectrum etc. Because signals are of random nature, the fusion usually uses Bayesian statistics with Kalman filter [11] as a typical representative. Results of higher-level fusion are statements (declarations) about instantaneous context, saying for instance that „ in the azimuthal angle "a" at the distance" d" is seen a small pond". In general, the signal fusion runs at the lower levels while the symbolic fusion runs at the higher levels. While a typical means used in signal fusion is Kalman filtering, a typical means used at higher levels is either Dempster –Shafer theory of evidence [12-14] or fuzzy logic [15].

It is important to note that results of the fusion (at all levels) are not only the estimated values (numeric or symbolic) but also corresponding *certainty values*. For instance, the result of the Kalman filtering is an estimate of the mean value and the variance of the mean is used as a measure of certainty. Contrary to this, in case of Dempster-Shafer evidence theory the output is a symbolic value (a statement), which is supplemented by its *belief value*. Finally, in the case of fuzzy fusion, the output is the consequent part of the fuzzy rule and the *degree of fulfillment* (firing strength) is a measure of certainty.

The higher level fusion is related to more sophisticated procedures of notion identification, i.e. "what was observed" and "what it means to have observed that". The higher level is a domain for application of *possibilistic approaches*, which can directly handle symbolic quantities, e.g. propositions. As indicated above, the proposition is accompanied by its certainty value (score), which expresses how certain the sensor is about its estimation of the measurand. Higher-level fusion is based either on *Bayesian statistics* (not mentioned here) or on the possibilistic approaches, like Dempster-Shafer evidence theory and fuzzy set theory. In what follows the two approaches are briefly described.

In the *Dempster-Shafer* approach all mutually exclusive declarations (propositions, hypotheses) are enumerated in the set T={$E_{1,}$ $E_{2...}$ E_n} called *frame of discernment*. Certainties are modelled as beliefs in one or more declarations or even ignorance. The *certainty value* "m" (called "*believe mass*" or "*basic probability assignment" - bpa*) [13, 14] can be assigned not only to a single declaration about the event (E) but also to the disjunction of events (E_1 OR E_2), or even to all subsets of T. Clearly, the set ? of all subsets of the T (termed as "*power set*") has 2^n elements. For instance, if T contains declarations about two dangerous contexts or events $E_{1,}$ E2 then ? ={E1, E2, (E_1 OR E_2), Ø} where Ø stands for an unknown event (ignorance)., With the power set ? defined, each sensor S_i would contribute by assigning it's belief "m_i "over all entries of ? . It is worth noting, that the term "sensor" does not mean only an instrument that measures simple physical quantities, but also more complex features like a dangerously discharged battery, visual or acoustic images, grade of the darkness, presence of people etc.

The inference engine then searches for the *total evidence* that supports the existence of event E, denoted "*believe (E)"* or *Bel (E).* The Bel (E) is given by the sum of the bpa's assigned to the set of those subsets B of the power set, which form a part of E, that is

$$Bel(E) = \sum_{\forall B, B \subseteq E} m(B)$$

The final belief in the existence of the event E (e.g. appearance of a dangerous context) is obtained by (recursive) application of *Dempster's rule of combination.* Let us suppose two sensors S_1 and S_2 with the respective bpa's *m1* and $m_{2.}$ The Dempster's rule of combination says

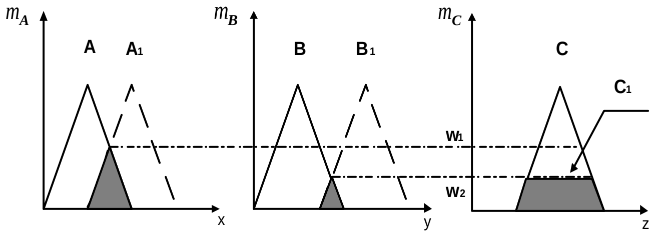
$$Bel(E) = \frac{\sum_{\forall \{B,C\} \in \Lambda : B \cap C = E} m_1(B) \cdot m_2(C)}{1 - \sum_{\forall \{B,C\} \in \Lambda : B \cap C = \emptyset} m_1(B) \cdot m_2(C)}$$

By definition the m(Ø) is set to zero.

In the *fuzzy approach*, the fusion process runs in two steps. The sensed signal values are first granulated in a process known as fuzzification. The measure of the certainty is expressed through corresponding values of membership functions. Within the second step runs decision-making, what is in essence a fusion of the statements. This is done by the *generalized modus ponens* (GMP) what is a generalized form of, in Boolean logic commonly used, *modus ponens*. Based on the GMP the fusion works as follows:

Declaration: *x is A_1 and y is B_1*
Fusion rule: *if x is A and y is B then* z is C
Conclusion: *z is C1*

The A1,B1,C1, A,B,C are fuzzy sets and crisp sets respectively. The A_1 is “close” to A, and B_1 is “close” to B Let w_1, w_2 are degrees of the matching between A and A_1, and B and B_1 respectively. Then $w_1 \wedge w2$ is the degree of fulfilment of the fuzzy rule, by which is the output membership function μ_c clipped, as could be seen in Fig 2.



$$m_{C1} = \boldsymbol{w_1} \grave{U} \boldsymbol{w_2} \grave{U} m_c$$

Figure 2: Fuzzy fusion by using the generalized modus ponens

Fuzzy logic offers the same advantages as DS theory, whereas it does not suffer from the exponential increase in complexity. Advantages of using fuzzy logic in data fusion are that it allows the capture of knowledge from a human expert in a very intuitive manner. That is why the fuzzy fusion was used in the development of a fault detection and classification system for a walking machine. In particular, sensor signals were first pre-processed by a set of fuzzy rules and then sent to the learning classifier, as described below. Preprocessing is not described in this paper.

3.0 CONTEXT AWARENESS AND FAULT DETECTION IN A WALKING MACHINE

Due to the extensive use of complex mechanical components like arms, legs, actuators, gears, clutches, grippers etc., the machine’s mechanical parts suffer from significantly higher fault rates than pure electric and electronic circuitry. The faults are detected, identified and classified in accordance with their criticality. Based on the results of classification appropriate measures are taken in order to prevent the system from failure. Potential faults should be detected sufficiently soon so as to avoid a fatal failure. In other words, the system should be able to anticipate possible faults on the basis of pathologic behaviors. In this view the *novelty detection* becomes necessary. Imminent failures are often manifested through the declined values of system parameters and variables or their fused complexes. An idea is to identify any deviation from normal behaviour. The most probable faults are expected to appear in the locomotion system dynamics; therefore they were detected and classified by using the bank of models tuned to a particular fault.

The *neural classifiers* are the most powerful means due to their learning ability. They can classify even noisy and sparsely populated sets of measured values. Main reason for using neural network (NN) instead of the traditional statistical means is in that the NN classifiers make weaker assumptions concerning the shape of statistical distribution of the input patterns. Another motivation for their use is need to detect the new and unexpected faulty contexts (problem of novelty detection). This can be achieved by unsupervised learning. A serious problem with NN classification is that the NN should preserve previously learned patterns (stability) while keeping its ability to learn new patterns (plasticity). This phenomenon is known as *stability-plasticity dilemma.* An elegant solution to this problem provides a family of the neural networks based on the *“adaptive resonance theory”* (ART), developed by Grossberg and Carpenter [16]. The ART family of self-organizing networks with competitive learning comprises network architectures,

which are able to cluster input patterns based on a given measure of similarity. In particular, the ART1 network, which was used in the experiment, allows the incremental learning of prototypes, rather then instantaneous input exemplars. That is because a cluster of similar inputs is updated using information from input pattern. In this way the cluster preserves main features of already accepted input patterns.
The ART1 depicted in Fig.3, consists of two fully connected layers. The comparison layer F_1 accepts the input pattern and through the button-up weighted connections b_{ij} (initially set to one) sends it to clustering layer F_2. (A binary coded input pattern is a concatenation of particular sensor outputs and the flags of the communication errors). Due to the competition based on the *winner-takes-all* paradigm, which is ensured by the negative lateral and positive own feedback (+1), the neuron in the F_2 layer what receives the highest button-up activity (let it be the i-th neuron) is declared a winner. Its output is set to unit value and projected back to F_1 through the top-down weights t_{ij}. If the similarity between the projected winner and the input pattern is greater then an *apriori* given value of the *vigilance ?*, so called *resonance* occurs, and weights t_{ij}, b_{ij} are (by different way) modified in such a way that they are moved closer to the input pattern. This is a learning step.

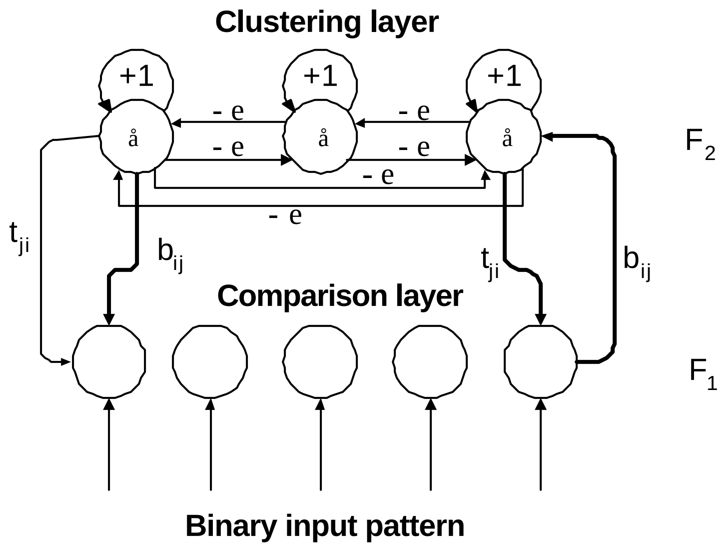


Figure 3: The ART-1 network

If the resonance does not occur, the winner is disqualified and the process searches for the second best matching neuron, which is then submitted to the vigilance test. Searching repeats until either vigilance test is passed or no more neurons are available for testing. It is just the chosen value of the vigilance *"?"* what calibrates how much novelty the neural network can tolerate before it clusters and classifies an input pattern into particular class. The experiments have shown that the ART-1 is fully justified for using as a means for the novelty detection and classification of patterns of the behaviors.

4.0 RESULTS OF EXPERIMENT

Efficiency of the developed neural classifier was verified by simulation as well as by experimentation with the walking machine developed for these purposes. The simplest and most evident faults like those related to control sequences that control the movement of joints and legs or the faults appearing during switching between particular gaits of walking were detected and classified by using the deterministic automaton developed for this purpose. It was possible due to the fact that these faults manifest themselves through the total fallouts of particular sensor signals.

More complex faults may be caused by increased friction in bearings, slipping or dragging clutches, lack of lubrication or a partial loss of energy delivered to particular joints. Another kind of these faults may be caused by incorrect coordination of legs due to improper timing (e.g. fall out of phase leading to time lag in the leg movement). Malfunctions of this kind may remain hidden for long time and, what is worst; they may gradually lead to serious failures, like destruction of bearings or drives what jeopardize the walking stability. Such faults are commonly manifested through abnormal trajectories of the joint torques or forces. Therefore, the learning neural classifier was designed just for the task of detection and classification of any abnormal joint torques.

In order to teach the neural network to classify abnormal torques, the legs dynamic was simulated in Toolbox SIMMECHANICS (a part of Simulink toolbox in MATLAB, oriented towards simulation of mechanical systems, including actuators and sensors.

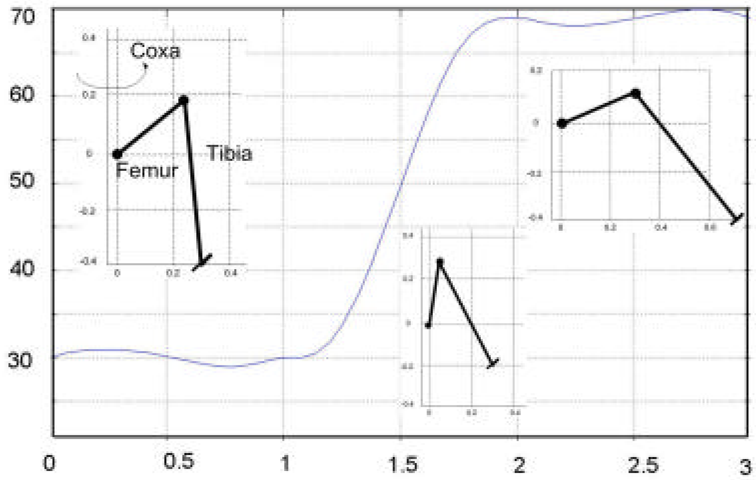


Figure 4: Time course of the normal torque in the femur joint

The leg can be either in a stance state, when it supports the body or in a swing state, when it moves in air to the position where it can begin a new stance. A time-course of the normal (faultless) torque exerted in a femur joint is shown in Fig.4 One complete step cycle is performed in three phases, each lasting one second. As seen from the Fig.4, these three phases can be easily observed from the graph of the torque-time dependence. Particular phases are supplemented with the imbedded simple sub-figures depicting the leg configuration corresponding to the particular phase. During the first phase the leg remains in a flexed configuration in the stance. The femur joint exerts torque about 30Nm, maintaining an attitude of the body. The second phase starts at one second. The leg is uncoupled from the ground and starts its swing movement in a direction of walking. While the torque exerted in the femur joint raises the leg, the coxa joint rotates the leg about the vertical axis and the tibia joint extends the leg. When reaching the highest position and maximum extension the leg ends its second phase. At this time instant the femur joint exerts maximum torque of about 70 Nm. Soon after the third second the femur torque slightly decreases so as to make the foot go down until it reaches the ground. At this moment (at about the fourth second) the leg is entering into its stance state again, and supports the body.

The neural network was first taught to learn the above-described normal torque. As a result, the neural network appointed this normal torque course as the centre of a receptive field of the cluster of all "approximately normal" torque courses (torque patterns). This is done by adaptation of the bottom-up weights of the winning neuron (the one located at most left position in the layer $F_{2)}$. From this time on, the unit value of this neuron will indicate that the current input pattern belongs to the cluster of "approximately normal" torque courses. Then a training list, i.e. a series of faulty torque patterns, generated by Simmechanics Toolbox, is repeatedly presented and clustered in accordance with their similarity. The experimental results have shown that the learning task is accomplished (the weights reach their steady values), after presentation about 5 or 6 epochs. After learning the neural network becomes able to classify successfully any other set of faulty torque time courses.

5.0 CONCLUSION

The secret behind the intelligence of artificial systems springs from successive generalization of sensor information into information chunks or "granules". Then the inferential process runs over (overlapping) information granules. Due to information granulation the system becomes robust with respect to imprecision, uncertainties, and partial truth. An intelligent machine, which is able to operate autonomously in an unknown environment, is a particular instantiation of the intelligent system. To this end, sensor data must be fused into information-rich patterns, which are further clustered and classified into classes corresponding to various contexts. Due to the classification of the contextual information the walking machine is able to distinguish between normal and erroneous behaviors. The methodology described above was used in the development of the learning neural fault detection and classification system of a walking machine. The ART1 neural network showed to be a very flexible and reliable means of detection any novelties appearing in the normal behaviour, i.e. novelties in the behaviour of a walking machine. Contrary to statistical approaches there is no need to specify number of classes in advance. Based on a chosen value of the vigilance ?, the network classifies input patterns into so many classes, how many is required for separation of dissimilar input patterns. In the more complex cases the system can detect and classified even contexts i.e. a composite state of the machine together with its environment.

6.0 REFERENCES

[1] Fu, K.S. (1971). *Learning Control Systems and Intelligent Control Systems: An Intersection of Artificial Intelligence and Automatic Control.* IEEE Trans. AC. (12), No 2, 17-21.

[1] Meystel, A., Messina, E. (2000). *The Challenge of Intelligent Systems.* Proc. of IEEE Int. Symp. on Intelligent Systems Rio Patras, Greece.

[2] Nauck, D.; Klawonn, F., Kruse R. (1997). *Foundations of Neuro-Fuzzy Systems*, 1st Edition, Chichester.

[3] Jang, J.S.R. & Sun,E.C.T., Mituzami E. (1997). *Neuro-fuzzy and Soft Computing* 1st Edition, Prentice Hall, Inc. Upper Saddle River.

[4] Zadeh L.A. (2001). *A New Direction in AI. Toward a Computational Theory of Perception*. AI Magazine, Spring 2001, 73-84.

[5] Brooks R.A. (1986). *Achieving Artificial Intelligence Through Building Robots.* AI Memo 899, MIT, Art.Intelligence Lab.

[6] Brooks R.A. (1989). *A Robot that Walks: Emergent Behaviours from a Carefully Evolved Network.* Proc. IEEE Conf.on Robotics and Automation, 692-696.

[7] Vitko A., Markusek J. & Jurišica L. (2001) *Unified Simulation Environment for Learning Navigation of a Robot Operating in Unknovn Terrain*. Proc.of CLAWAR, Karlsruhe, Germany, 435-441.

[8] Mataric M. J. (1998). *Behaviour-Based Robotics as a Tool for Synthesis of Artificial Behaviour and Analysis of Natural Behavior.* Trends in Cognitice Sciences, No. 3, 82-87.

[9] Saffiotti A. (1997). *The Use of Fuzzy Logic in Autonomous Robot Navigation*: a catalogue raisonné Technica Report TR/IRIDIA/, Univ.Libre de Bruxelles, 97-126.

[10] Maybeck, P.S. (1974). *Applied Optimal Estimation-Kalman Filter Design and Implementation.* Notes of a continuing education course offered by the Air Force Institute of Technology, Wright-Patterson AFB, Ohio, since Dec. 1974.

[11] *Sensor Data Fusion* www.wins.uva.nl/˜arnoud/education/OOAS/fwi/Cap9. pdf

[12] Wu, H. et al. (2002). Sensor Fusion Using Dempster –Shafer Theory. *IEEE Instrumentation and Measurement Technology Conf., Anchorage, AK, USA,* May, 21-23.

[13] Klein.L.A. (2001). *Sensor Technologies and Data Requirement for ITS.* 2nd edition, Artech House.

[14] Mendel J.M.,T. (1995). *Fuzzy Logic System for Engineering: A Tutorial.* Proc. of the IEEE, Vol.83, No. 3, 345-377.

[15] Carpenter. G.A, Grossberg. S. (1987). *A massively parallel Architecture for a Self-organizing Neural Pattern Recognition Machine,* Comp. Vision, Graphics and Image Proc. Vol. 37, 54-115.

[16] Baraldi, A., Alpaydin. (2002) *Constructive feedforward ART Clustering Networks*. IEEE Trans on Neural Networks, Vol. 13, 645-661.